Tangible interaction toolkits for designers

Camille Moussette
Umeå Institute of Design
Umeå University, Umeå, SE-901 87
Sweden
camille@guchmu.com

ABSTRACT

As new products and systems grow in complexity and embed new technologies, designers face new challenges for adequately pursuing their prototyping activities. Tangible User Interface toolkits provide an easier development environment for rapidly exploring and prototyping functional physical interactive devices. Such toolkits range from friendly toy-like building blocks to advanced wireless microchips, and provide designers the ability to explore and play with bits and atoms like never before. This paper briefly reviews seven electronic toolkits and discusses their present and future use in industrial and interaction design projects.

Keywords

Tangible, interaction, toolkit, prototypes, mock-up, exploration, interface, TUI.

1. INTRODUCTION

Our interactions with the physical world are rich, complex and non-trivial. We use, manipulate and interact both intentionally and passively with hundred of products and services on a daily basis. A growing number of the products and services that we use are neither only physical (i.e. toothbrush) nor only virtual (i.e. website), but present complex interfaces mixing both tangible and intangible attributes.

Over the years, designers have developed numerous skills and methods for building representations, mockups, models and prototypes to inform both the design process and the design decisions. Prototyping is now viewed as an essential path to innovation and as a key activity in the creative design process [7][10][11]. Prototypes, from paper sketches to fully functional models, serve many purposes and goals. They are temporary representations that “look like”, “behave like” and “work like” used to explore and communicate design ideas [1].

As new products and systems grow in complexity and embed new technologies (pervasive computing), designers face new challenges. Designing and prototyping such products and interfaces becomes non-trivial [5]. The required knowledge usually spans various disciplines as both the physical and the digital worlds have to seamlessly integrate into one. Unfortunately, not all projects can afford a full interdisciplinary team of designers, sociologists, programmers and engineers.

Tangible interface toolkits aim at democratizing and simplifying some prototyping activities and techniques often used in design practice. They provide simplified accesses and controls to areas usually inaccessible to designers: low-level hardware and programming, embedded electronics, wireless communication, etc. Designers, using such tools, can rapidly develop prototypes that were out of their realm before. With minimal knowledge and training, they can now explore and play with bits and atoms like never before. [9][12]

This article presents a short review of the major electronic toolkits commercially available to designers (as of January 2007). It briefly details their respective features, characteristics, strengths and weaknesses. A discussion follows in order to see what kind of trade-offs or compromises they represent. Some successful use cases are then presented before discussing future development in this area: what is missing, what is coming.

2. TOOLKITS

The following toolkits represent in the author’s experience and best knowledge a good selection of the products commercially available to individuals, students, schools and businesses in the field of product and interaction design. Many more tools and systems probably exist, in other related markets and throughout research groups around the globe. The presentation and description of the various toolkits provided in this article are very concise. A thorough presentation and analysis is available on the author’s website [16].

2.1 Phidgets

Phidgets are easy to use building blocks for interfacing the physical and the virtual worlds. This system arose out of a research project at the University of Calgary in Canada. The philosophy behind it is “… just as widgets make GUIs easy to develop, so could phidgets make the new generation of physical user interfaces easy to develop.“ Modules are plug and play and don’t require soldering of electronic components. The system has an extensive library of Application Programming Interfaces (APIs) and can be used with a large number of applications, even with other toolkits in some cases.

2.2 BasicStamps - BasicX - BasicAtom

This review will focus on the BasicStamps from Parallax Inc. as they are the more popular. Many other manufacturers produce very similar chips and development modules.

Parallax modules started in 1992. Now they offer 8 different types of BASIC Stamp modules with varying capabilities (speed, memory, power consumption). The modules are actually a PIC microcontroller nicely packaged (supporting circuitry) for easy programming and interfacing.

2.3 Make controller

A new generation of open source hardware platform, successor to the Teleo modular kits. This system is a collaboration with the MAKE magazine that specifically embrace the do it yourself (DIY) subculture. This kit is targeted at enthusiasts and hobbyists. It offers extensive features and interfaces.

2.4 I-CubeX

This proprietary system is based on the MIDI communication protocol and offers modular components covering a large field of applications. The components (sensors, actuators, digitizing modules and such) are of very high quality and easy to connect (almost configuration-free), but this comes at a cost. A basic kit costs around 700 USD while typical modules range from 40 to 100 USD. This system is clearly intended for production or serious integration, with its high price and neat little black boxes. Nevertheless, the possibilities offered by this system are quite large. Thanks to the MIDI protocol, it is fairly straightforward to integrate the I-CubeX with other systems like musical instruments, stage equipments, light modules, video and audio units.

2.5 Wiring and Arduino

“Arduino is an open-source physical computing platform based on a simple i/o board, and a development environment for writing Arduino software.” (from the Arduino website). The philosophy of this platform is to have products that are open source, easy to use and cheap to produce. Wiring is a similar platform but features a more sophisticated I/O board and development environment.

2.6 Lego Mindstorms

Robot oriented computer-controlled kit by LEGO. The system offers various sensors and motors linked to a central programmable unit.

2.7 Barebone - custom solutions

Microcontrollers are also available in “barebone” configuration. It is up to the designer/programmer to build the supporting circuitry and necessary code. Some common configurations (dedicated to specific tasks) are available as ready to use modules by some vendors, i.e. LCD screens, LEDs circuits, light sensors, etc.

3. DISCUSSION

All of the toolkits reviewed here feature a microcontroller at their core, with some sort of circuitry for connecting and communicating with external devices (power supply, sensors, outputs, etc.). All systems can also interface with a computer for processing heavier calculation, for reprogramming or reconfiguring the system, or both. Each system has specific capabilities and characteristics. It is impossible to define an overall better system. Every project and prototyping activity has different needs and requirements. For every situation, there will be trade-offs or compromises somehow. This diagram depicts one set of attributes that could be considered:

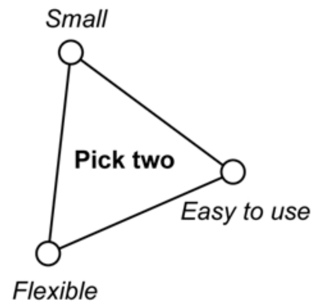

Figure 1. A typical trade-offs representation

Each toolkit was designed for specific purposes and markets. Most of them were not developed solely and specifically for designers. They might not completely fulfill your particular needs directly. Nevertheless, they can be extremely valuable tools in some instances. In order to see a global view on how each system compares and relates to each other I have mapped them along two axis: the technical level required in order to use the system and the price.

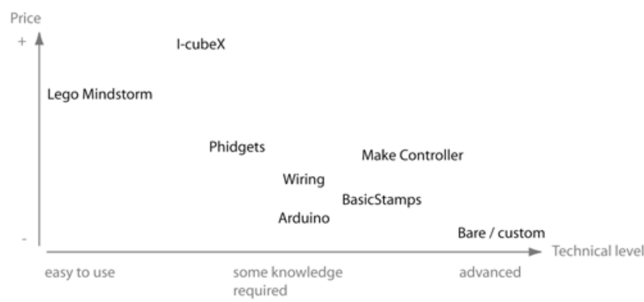

Figure 2. 2D mapping of the various toolkits.

It is important to note that these toolkits represent only a fraction of the tools and skill needed for electronic prototyping activities. Buttons, switches, power supply, wires, raw materials, lights, pumps, speakers and many other devices need to be considered and put into place for building any kind of projects. The toolkits offer simplified platforms or shortcuts to complex development processes. Designers should evaluate carefully their needs and capabilities. In some cases, it might be beneficial to get assistance from a hardware engineer or other experts. For some situations, partially or totally relying on non-technological prototyping techniques like Wizard of Oz or paper-based solutions could be far more appropriate, efficient and less time-consuming.

4. USE CASES

4.1 Wintson, the animated teddy bear

The goal of this project was to give life to an unanimated object. Using Phidgets and Flash programming, the two designers were able to bring a Teddy Bear to life under four days of work. It is important to note that the designers had previous experience in programming and electronics before starting this project. An animated skeleton was built using servos, metal wire, an old desktop speaker and hot glue. Switches and lights were added for extra interaction, before dressing up the internal parts with the bear suit. Phidgets modules were used for controlling the servos and lights (eyes) and sensing user's touch around his belly. A servo board plus one interface kit were used simultaneously. Additional input modules were used during development for programming the movement sequences of the arms and head. Switching between recording mode and playback mode was accomplished via software (Flash) and didn't require unplugging or swapping wires.

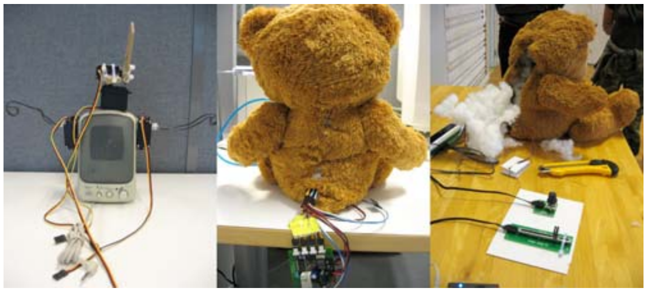

Figure 3. 2D Skeleton of the animated bear. Final mockup with connections coming out from his back. Puppeteer's control modules for animating the beast.

The major problem encountered with the toolkit during this project was the servo board and servo motors. First, the servo board tended to overheat and became instable after 25-30 minutes of use. Second, as you plug and unplug the servo board, the unit goes into in a initialization routine and set the servos to a default position. This behavior forced the servos to work passed their physical limit (amplitude limited by the bear suit) and ultimately dislodge from the skeleton. It was also impossible to override the power-on sequence via software to avoid this behavior.

Nevertheless, the plug and play nature of the Phidgets items, along with its easy interfacing capabilities with Actionscript provided a very capable platform for rapidly developing this project. The Phidgets system has its inevitable weaknesses (large number of wires, permanent connection to computer) but a project like this simply wouldn't have been possible to build otherwise, especially in such a short period of time.

4.2 InVivo, ecological electrical kettle

Figure 4. Prototype of the on-demand kettle with BS2 microcontroller hidden into it.

A BasicStamp (BS2) module was use to implement and test the usage sequence for this new type of hot water dispenser. After powering on the device, the light progression across the LED depicts the status of the machine: off, warming up, ready to serve. Once the heating element is ready, the rotating valve pulses green. The relatively simple prototype with the working interface made it possible to experience it live instead of explaining how it would be. The feedback and understanding of the concept from the public and the tutors was much richer in this regard. It was possible to implement the interface using a BasicStamp toolkit, simple switches and LEDs, all together programmed with about 40 lines of code. It took less than day of work to plan, buy the required pieces and assemble the demo interface.

5. FUTURE DEVELOPMENT

As Moore' law suggests, the components found in these toolkits will rapidly get smaller, faster and cheaper over time. Unfortunately, power and price of these components are not the limiting factors in today's systems. As we have seen with the barebones solutions, powerful microcontrollers only cost 4 or 5 dollars. Raw electronics sensors and components are also quite cheap already.

The values of electronic toolkits reside in the fact that they offer an unique combination of microcontroller, supporting circuitry, and development environment (hardware and software). They make the complicated accessible, to some extent.

Any designer who has use such toolkit will tell you that there is still a lot of room for improvements. The complicated mixture of software and hardware often results in a lot of problems and frustration. Troubleshooting and diagnosis activities can quickly become overwhelming, for novice and expert alike.

Newer tools and systems are becoming more "intelligent" in this regard. They now have built-in diagnosis and sensing capabilities that can greatly reduce troubleshooting and debugging activities. NADA [8], a software component, proposes for example interactive help content on the computer. Its documentation and help pages change according to what is plugged in on the hardware devices.

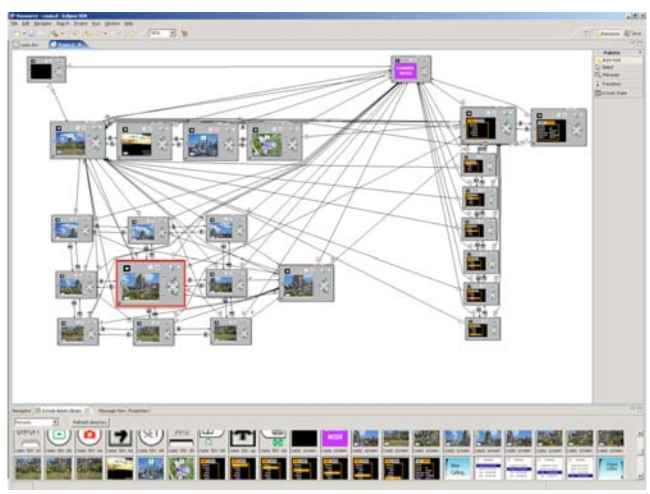

Figure 5. D.tools visual editor for arranging and linking states of the hardware nodes.

D.tools [4], a new hardware and software system from a research group at Stanford University, also goes in the same direction. It proposes a tighter coupling between the hardware modules and a visual software editor. Each hardware nodes can be tracked, configured and edited via software at all time. The system looks really promising, in particular for prototyping and evaluating devices with complex data (images, video, sound clips).

Over the last few years, research projects related to ubiquitous computing have flourished and produced a large breed of experimental miniature devices [2][3]. These networked, sensor-packed and quasi-autonomous systems will undeniably push the limits of how we build, prototype and use electronics devices. Small, wireless and always-on sensors will probably find good use in the designer's prototyping toolbox. Logging and tracking user's actions and behaviors for development purposes suddenly becomes trivial.

No matter how small or fast the components are (or will be), they should properly fit the needs and workflow habits of their target users. Where the first generation of toolkits targeted engineers and technically inclined people, some newer systems like the Calder toolkit [6] focus specifically on the needs of product and interaction designers. This new perspective will change the way these toolkits are developed and used.

6. CONCLUSION

Prototyping interactive products and tangible interfaces generally requires substantial time and effort by individuals with highly specialized skills and tools. This situation limits the designers' ability to approach and successfully work on such projects. To help overcome this difficulty, designers have recently turned their attention to some electronic toolkits. These systems provide simplified and far more accessible development environments for product and interaction designers to prototypes their work.

Seven different electronics toolkits particularly well suited for designers were reviewed in this article. I am sure a lot more exist and will be developed in the near future. Each system presents their own set of capabilities, strengths and weaknesses.

As demonstrated in the two projects, the use of a toolkit provided designers with a very good way to rapidly explore and prototype functional physical interactive devices.

Twenty years ago, designers had very limited tools and assets to deal with Graphical User Interfaces (GUIs). Over the years, interface tools and toolkits have progressed to a point where designers can now develop high-quality, iterative and user-centered GUIs effectively. History seems to repeat itself as the situation with tangible interfaces now echoes the past experiences of the GUI community.

7. REFERENCES

[1] Buchenau, M. and Suri, J. F. 2000. Experience prototyping. In Proceedings of the Conference on Designing interactive Systems: Processes, Practices, Methods, and Techniques (New York City, New York, United States, August 17 - 19, 2000). D. Boyarski and W. A. Kellogg, Eds. DIS '00. ACM Press, New York, NY, 424-433.

[2] Decker, C., Krohn, A., Beigl, M., and Zimmer, T. 2005. The particle computer system. In Proceedings of the 4th international Symposium on information Processing in Sensor Networks (Los Angeles, California, April 24 - 27, 2005). Information Processing In Sensor Networks. IEEE Press, Piscataway, NJ, 62. http://particle.teco.edu/upart

[3] Gellersen, H. 2005. Smart-Its: computers for artifacts in the physical world. Commun. ACM 48, 3 (Mar. 2005), 66. http://smart-its.teco.edu

[4] Hartmann, B., Klemmer, S. R., Bernstein, M., Abdulla, L., Burr, B., Robinson-Mosher, A., and Gee, J. 2006. Reflective physical prototyping through integrated design, test, and analysis. In Proceedings of the 19th Annual ACM Symposium on User interface Software and Technology (Montreux, Switzerland, October 15 - 18, 2006). UIST '06. ACM Press, New York, NY, 299-308. http://hci.stanford.edu/research/dtools

[5] Hornecker, E. Tangible Interfaces and Graspable Interfaces. http://www.ehornecker.de/Tangibles.html

[6] Lee, J. C., Avrahami, D., Hudson, S. E., Forlizzi, J., Dietz, P. H., and Leigh, D. 2004. The calder toolkit: wired and wireless components for rapidly prototyping interactive devices. In Proceedings of the 2004 Conference on Designing interactive Systems: Processes, Practices, Methods, and Techniques (Cambridge, MA, USA, August 01 - 04, 2004). DIS '04. ACM Press, New York, NY, 167-175.

[7] Moggridge, B. 2006 Designing interactions. MIT press, USA. 701-735.

[8] NADA http://www.sketchtools.com

[9] O'Sullivan, D. & Igoe, T. 2004. Physical Computing: Sensing and Controlling the Physical World with Computers. Thomson Course Technology PTR. USA.

[10] Preece, J., Rogers, Y., Sharp, H., Benyon, D., Holland, S., and Carey, T. 2002 Interaction Design : Beyond Human-Computer Interaction. John Wiley & Sons Inc.169, 275.

[11] Saffer. D. 2006 Designing for Interaction: Creating Smart Applications and Clever Devices. New Riders, USA. 114-119.

[12] Ullmer, B. & Ishii, H. 2001. Emerging Frameworks for Tangible User Interfaces. In Human Computer Interaction in the New Millenium, John M. Carnoll (Ed.), Reading, MA: Addison-Wesley, 579--601.

8. FURTHER READINGS

[13] MIT Media Lab, Tangible media group http://tangible.media.mit.edu

[14] Physical Computing at ITP http://itp.nyu.edu/physcomp/Intro/HomePage

[15] Barragán, H. 2004. Wiring: Prototyping Physical Interaction Design. Interaction Design Institute Ivrea. Italy

[16] Moussette, C. 2007. Tangible interaction toolkits for designers. http://www.guchmu.com/publications/toolkits